

REMARKS

Reconsideration and allowance of above referenced application are respectfully requested. Upon entry of this amendment, claims 1, 2, 4-45, 47, 49-66, 68-71, 73, 74, 76-81 and 83-105 will be pending in this application.

Applicant thanks the Examiner for the time spent during the telephonic interview conducted on July 16, 2001.

Applicant also thanks the Examiner for his indication that claims 15, 24, 47, 51, 63, and 78 are allowed, and that claims 4-15, 17-19, 22-23, 25-45, 49-50, 52-53 57-62, 68, 70 and 83 contain allowable subject matter.

With respect to the informalities noted in claims 69, and 76-77, these have been corrected.

Claims 5-15, 17-19, 21-23, 25-45, 52-53, 57-62, 64, 66, 68-70, 72-74. 76, and 77 stand rejected under 35 USC 112, second paragraph, as being indefinite because it is unclear whether they are reciting the method of the base claim, or a new or another method. Since each of these claims are dependent upon a base claim, of course they are reciting the method of the base claim, and accordingly applicant's attorney does not understand the Examiner's reasons for continuing to bring this rejection. Nonetheless, in order to expedite prosecution of this application, the requested change of the word "A" to "The" in each of these claims has been made. It is noted that this change is not one that is being made for reason related to patentability, but only a matter of form.

With respect to the rejection of claim 7, the antecedent for "the maximum size" is in claim 5.

With respect to claims 4, the word "portion" had only been used to modify the new data stream, and so it is submitted that the recitation set forth was clear. Nonetheless, in order to expedite prosecution of this application, this has been made even clearer.

With respect to claim 22, what m and n are is very clear, and, therefore, it is submitted that the claim is definite and readily understandable without a recitation of ranges, as suggested by the Examiner.

With respect to claim 54, as this claim is an independent claim, the Examiner's reference to claim 51 is not understood, and it is submitted that this claim is definite and readily understandable.

With respect to claim 60, the limitation "said frame" refers to the antecedent basis "a frame" in that same claim. Though claim 60 is dependent on claim 51, "a frame" is first referenced in claim 60 and it is submitted claim 60 is definite and readily understandable.

With respect to claim 66, the dependency error has been corrected, and it is submitted that this claim is definite and readily understandable.

Claim rejections have also been made to claims 1-2, 20-21, 54, 71-74 and 79-81 under 35 USC 103(a) as being unpatentable over U.S. Patent No. 6,058,109 to Lechleider in view of U.S. Patent No. 6,137,834 to Wine Porter et al.

In order to illustrate the advantageous features of the present invention, Applicant draws to the attention of the Examiner that the assignee of the present invention, Optibase, formerly Viewgraphics, received the "Editors Pick of Show" award at the annual National Association of Broadcasters (NAB) convention in 1998 for the MediaSplice prototype product shown that implemented the present invention. This provides one indication that the present invention was not obvious to one of ordinary skill in the art at the time the invention was made.

Independent claims 1, 20, 54, 71, and 79, as explained in the previously submitted amendment, when applicant was distinguishing over a different combination of references used by the Examiner to reject these claims. These claims each recite, in some form, delaying a first data stream portion, and accelerating a second data stream portion to make up for the delay time of the first data stream portion -- a combination that has been found to be very advantageous.

In hindsight, the Examiner now selects two patents that describe very different systems, and attempts to assert that the claimed inventions are obvious. Respectfully, that is not the case.

Initially, given the complex nature of the systems in each of Lechleider and Wine, one of ordinary skill in the art would not have been motivated to combine them --particularly in the manner suggested by the Examiner.

The Lechleider system, as discussed by the Examiner at page 5, lines 8-10, requires "the receiving system and the sending system [to] alternatively communicate the number of empty receive buffers and data bits to fill the buffers." In such a system that requires these back-channel communications, implementation of the "acceleration concept of the present invention" would just not be practical, since the time for acceleration may well be past by the time the back-channel communication signals were communicated.

Wine requires an "entrance indicium" that identifies the appropriate point of entrance to the stream, and an "exit indicium" that identifies an appropriate point of exit from the stream. This extra data requires specific encoding in order for the Wine invention to work. And, furthermore, since this data in Wine is coded in MPEG, it is not intended to work with a system that has back-channel

communication, as in Lechleider, since there is no back-channel communication in MPEG.

Accordingly, it is apparent that one of ordinary skill in the art would not have been motivated to combine Lechleider and Wine.

Further, while the Examiner has the benefit of applicant's specification and can conclusorily assert that it "would have been obvious" to combine Lechleider and Wine, it is again respectfully noted that for any such combination, the combination cannot destroy the essential characteristics of each. *In Re Gordon*, 733 F.2d 900, 221 USPQ 1125 (Fed. Cir. 1984). And Applicants respectfully assert that the Examiner's pending rejections have not fully considered the implications of this requirement as it pertains to the present inventions. In particular, the Lechleider invention only works on streams of data in a system that has back-channel communications capability. The Wine system requires a special encoding technique. Neither of these aspects is used by the other. How those essential characteristics of each can be maintained in a workable system has not been described by the Examiner --and reinforces that the Examiner is inappropriately using hindsight.

And even if they were combined, the present invention would not result.

Lechleider does not teach or suggest such acceleration, and nor does Wine.

The Examiner acknowledges that Lechleider does not, but asserts that Wine does. Wine, however, discloses a system that requires an "entrance indicium" that identifies the appropriate point of entrance to the stream, and an "exit indicium" that identifies an appropriate point of exit from the stream, as mentioned above. As taught by Wine at column 3, lines 22-24, "it is desirable to include as many in-points and out-points as possible in a stream to allow for maximum flexibility in splicing." Thus, when read in context with the passage cited at column 15, lines 8-17, it is apparent that the prevention of underflow that is discussed is not accelerating as is taught by the present invention, but rather properly aligning the exit indicium of one stream with the entrance indicium of another stream. The sentence at the end of this cited passage, which reads "[In other words, the next I-frame (the first frame of the to-stream) must be delivered to the encoder buffer before the buffer is emptied [.], " reinforces this conclusion.

Non-obvious advantages are also gained by the present invention, which include the ability to introduce a delay between encoding and transmission, to use existing hardware to implement the system, and to splice material that has already been encoded. Accordingly, it is respectfully submitted that the claimed combination is patentable.

Still furthermore, claim 1 also introduces that the acceleration substantially makes up for the

delay. That the amount of the acceleration is related to the amount of previously introduced delay is nowhere taught or suggested by the cited references. And similarly, claim 76 specifically recites deleting a number of null packets corresponding to the acceleration parameter recited therein, which is not disclosed or suggested by the cited references.

Claims 55 and 56 stand rejected under 35 USC 103 as being unpatentable over U.S. Patent No. 6,049,569 to Radha et al. in view of U.S. Patent No. 5,864,682 to Porter et al.

Independent claims 55 and 56 are directed to a method for and apparatus that prepares a digitally encoded data stream for splicing. As explicitly recited independent claims 55 and 56, this includes the step of and structure for, respectively, closing an initial open group of pictures (GOP).

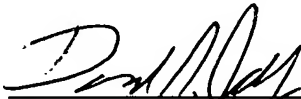
The Examiner acknowledges that Radha does not disclose the feature of closing an initial group of pictures if the new data stream includes an initial open group of pictures. The Examiner asserts that the "adding prefix and suffix data" in Porter to "ensure that the new video file conforms to the desired format" corresponds to this feature. Respectfully it does not. Adding data as required by Porter does not correspond to closing an open group of pictures by removing frames within an initial open group of pictures in order to close the group. To make this more explicit, claims 55 and 56 have been amended to clarify that closing a GOP includes the removal of open frames, which clearly Porter does not do.

Further, new claims 84-105 have been added, and are also believed to contain patentably distinct subject matter.

In view of the above amendments and remarks, Applicant submits that the above-referenced application is in a condition for allowance and respectfully requests a Notice to that effect.

Respectfully submitted,

PILLSBURY WINTHROP, L.L.P.



David A. Jakopin, Reg. No. 32,995

Tel. No.: (650) 233-4790

Fax No.: (650) 233-4040

1600 Tysons Boulevard
McLean, Virginia 22102

APPENDIX SHOWING CHANGES TO SPECIFICATION AND CLAIMS

IN THE SPECIFICATION:

[Paragraph at page 50, line 12, to page 51, line 7]

The FIG. 12 flow chart illustrates how a preferred splicer according to the invention preferably aligns the start of decoder input buffer receipt of new data stream data with the start of the last buffer of the old data stream. Broadly stated, if the new data stream is an early data stream type (e.g. curve 1102b of FIG. 11), then splicer-N 270 delays transmission of (and equivalently, delays reception by a decoder of) the new data stream data. If instead, the new data stream is a late stream type (e.g. curve 1102c of FIG. 11), then splicer-N 270 accelerates transmission of (and equivalently, accelerates reception by a decoder of) new data stream data. More specifically, splicer-N 270 preferably delays the new data stream data by adding null packets prior to the start of new data stream transmission, or ~~{delays}~~ **[accelerates]** the new data stream by deleting null packets that have been encoded into the data stream. Splicer-N 270 further preferably stores a total number of added/deleted null packets for later use during splicing, as will be discussed further herein.

IN THE CLAIMS

Please cancel claim 72.

Please amend the following claims:

1. **(three times)** ~~{1. (twice)}~~ amended) A method for removing an overflow condition comprising the steps of:
 - detecting a first digitally encoded data stream portion causing said overflow condition;
 - delaying said first data stream portion for a delay time that prevents said overflow condition; and
 - accelerating a second data stream portion that ~~{is preceded by}~~ **[follows]** said first data stream portion to substantially make-up for said delay time.

2. ~~{{(twice)}}~~**[(three times)]** amended) An apparatus for removing an overflow condition comprising:

means for detecting a first digitally encoded data stream portion causing an overflow condition;

means for delaying said first data stream portion; and

means for accelerating a second data stream portion that ~~{is preceded by}~~ **[follows]** said first data stream portion.

4. ~~{{(twice)}}~~**[(twice)]** amended) A method for avoiding overflow of a decoder buffer comprising:

(a) determining a total amount of old data stream data that, if transmitted to said decoder buffer, would occupy said decoder buffer;

(b) adding, to said total amount, an amount of new data stream data to obtain a combined amount of data;

(c) testing if said combined amount of data would overflow said decoder buffer; and

(d) if overflow would occur, then causing a portion of the new data stream to be delayed by a delay amount corresponding to at least said overflow, if said portion **[of the new data stream]** were to be transmitted to said decoder buffer.

5. ~~{{(amended) A}}~~**[(twice amended) The]** method according to claim 4, wherein the step (a) of determining is preceded by determining a maximum size of said decoder buffer.

6. ~~{A}~~**[(amended) The]** method according to claim 5, wherein said maximum size is determined according to a buffer size parameter within the old data stream.

7. ~~{A}~~**[(amended) The]** method according to claim 5, wherein said maximum size of step (a) is determined according to a buffer size parameter within the new data stream.

8. ~~{{(amended) A}}~~**[(twice amended) The]** method according to claim 4, further comprising:

prior to testing of step (c), subtracting, from said total amount, an amount of old data stream data that, if transmitted, would be decoded by a decoder;

9. ~~{(amended) A}~~ **{(twice amended) The}** method according to claim 4, wherein said delay amount of step (d) is a function of an amount of data stream data by which said decoder buffer is overflowed within said portion of the new data stream.

10. ~~{(amended) A}~~ **{(twice amended) The}** method according to claim 4, wherein said delay amount of step (d) is a function of an amount of data stream data by which said decoder buffer is overflowed in a single instance of overflow within said portion of the new data stream.

17. ~~{(amended) A}~~ **{(twice amended) The}** method according to claim 16, wherein said second plurality of old data stream frame sizes of step (d) include all frames of the old data stream portion that will remain un-decoded when said new data stream frame will be received by the decoder, if the data stream portions are transmitted.

18. ~~{A}~~ **{(amended) The}** method according to claim 17, wherein the data stream portions are transmitted.

19. ~~{(amended) A}~~ **{(twice amended) The}** method according to claim 16, further comprising:

if overflow is found in step (f), then causing a transmission time of a portion of new data stream data including said new data stream frame to be delayed.

21. ~~{A}~~ **{(amended) The}** method according to claim 20, wherein said delay is caused by adding null packets to said new data stream portion.

23. ~~{A}~~ **{(amended) The}** method according to claim 22, wherein m equals 1504.

25. ~~{A}~~**[(amended) The]** method according to claim 24, wherein said modified new data stream timing reference of step (a) further corresponds with a timing gap between a first decoding time for decoding a last frame of the old data stream and a second decoding time for decoding a first frame of the new data stream.

26. ~~{A}~~**[(amended) The]** method according to claim 24, wherein determining said modified new data stream timing reference includes:

- (i) determining said current timing reference of the new data stream;
- (ii) determining a delay between said current timing reference and a current decoding time of a frame of the new data stream;
- (iii) determining a new decoding time of said frame of the new data stream that corresponds with a sum of said current decoding time and an inter-frame delay between a decoding time for decoding a last frame of the old data stream and a decoding time for decoding a first frame of the new data stream; and
- (iv) determining said modified new data stream timing reference as said new decoding time of step (iii) minus said delay of step (ii).

27. ~~{(amended) A}~~**[(twice amended) The]** method according to claim 24, wherein determining said modified new data stream timing reference includes:

- (i) determining a program clock reference of a first packet of said new data stream;
- (ii) determining a delay between transmission of a first sequence header of said new data stream and a first decode time stamp DTS of a first frame of said new data stream;
- (iii) determining a continuous DTS as a sum of said first DTS and an inter-frame delay; and
- (iv) determining a new data stream real-time transmit time as said continuous DTS of step (iii) minus said delay of step (ii).

28. ~~{(amended) A}~~[(twice amended) **The]** method according to claim 24, wherein said aligning in step (b) sets a start time for transmitting the portion of the new data stream that corresponds with a decoding time for decoding the portion of the old data stream.

29. ~~{(amended) A}~~[(twice amended) **The]** method according to claim 24, wherein said aligning in step (b) sets a start time for a decoder buffer to begin receiving the portion of the new data stream that corresponds with a decoding time for decoding the portion of the old data stream.

30. ~~{A}~~[(amended) **The]** method according to claim 24, further comprising:

(d) detecting a decoder buffer overflow condition that will result from said splicing, if the data streams are transmitted; and

(e) correcting said overflow condition.

31. ~~{(amended) A}~~[(twice amended) **The]** method according to claim 24, wherein said determining of step (a) is preceded by

(i) determining the splice-out point of the old data stream; and

(ii) determining the splice-in point of the new data stream.

32. ~~{A}~~[(amended) **The]** method according to claim 31, wherein step (ii) includes, if an initial frame of the new data stream is of a type that is ordinarily decoded with reference to decoding of a prior frame, then modifying the new data stream to remove said reference.

33. ~~{(amended) A}~~[(twice amended) **The]** method according to claim 32, wherein said frame type is selected from a group consisting of B-frames and P-frames, and wherein said step of modifying comprises closing an open group of pictures GOP.

34. ~~{A}~~[(amended) **The]** method according to claim 31, wherein said data streams include video and audio data, wherein step (a) includes determining a video splice-out point and an audio

splice-out point, and wherein step (b) includes determining a video splice-in point and an audio splice-in point.

35. ~~{A}~~[(**amended**) **The**] method according to claim 31, wherein said splice-out point of step (i) is determined within a user-selectable portion of the old data stream.

36. ~~{A}~~[(**amended**) **The**] method according to claim 31, wherein said splice-in point of step (ii) is determined within a user-selectable portion of the new data stream.

37. ~~{A}~~[(**amended**) **The**] method according to claim 31, wherein said splice-out point of step (i) is user-selectable.

38. ~~{A}~~[(**amended**) **The**] method according to claim 31, wherein said splice-in point of step (ii) is user-selectable.

39. ~~{A}~~[(**amended**) **The**] method according to claim 24, wherein step (a) is preceded by determining a first source for the old data stream and a second source for the new data stream.

40. ~~{(amended)-A}~~[(**twice amended**) **The**] method according to claim 39, wherein said first and second sources include source types selected from a group comprising a storage device, a satellite receiver, a cable receiver, a network, an audio source, a video source and an encoder.

41. ~~{A}~~[(**amended**) **The**] method according to claim 40, wherein said first source and said second source are of a same source type.

42. ~~{A}~~[(amended) The] method according to claim 24, wherein at least one of said data streams is MPEG encoded.

43. ~~{A}~~[(amended) The] method according to claim 24, wherein said splicing is accomplished in real-time.

44. ~~{A}~~[(amended) The] method according to claim 24, wherein step (a) is followed by transmitting a portion of the old data stream.

45. ~~{(amended) A}~~[(twice amended) The] method according to claim 24, wherein step (b) is followed by transmitting the portion of the new data stream.

49. ~~{(amended) A}~~[(twice amended) The] computer-readable storage medium according to claim 47, wherein the step of determining the splice-out point is preceded by:
creating at least one data storage structure for storing portions of said old and new data streams; and
storing portions of said old and new data streams in said at least one data storage structure.

50. ~~{A}~~[(amended) The] computer readable medium according to claim 49, wherein said at least one data storage location is located in memory of a host processing system.

52. ~~{(amended) A}~~[(twice amended) The] method according to claim 51, wherein said dependency of step (f) is an open GOP and wherein said modifying closes the open GOP.

53. ~~{(amended) A}~~[(twice amended) The] method according to claim 51, further comprising:

checking for overflow of said decoder buffer; and
if overflow is found, then removing said overflow.

55. ~~{(}~~**[(twice] amended)** A method for preparing a digitally encoded data stream for splicing, comprising:

(a) determining a splice-in point of a new data stream; and

(b) closing an initial open group of pictures GOP of the new data stream, if the new data stream includes an initial open GOP[, **wherein the step of closing the GOP includes the step of removing open frames.**]~~{.}~~

~~{56. (}~~

[56. (twice] amended) A splicer for splicing digitally encoded data streams including an old data stream and a new data stream, comprising:

(a) means for determining a splice-in point of the new data stream; and

(b) means for closing an open group of pictures GOP of the new data stream, if the new data stream includes the open GOP[, **wherein the step of closing the GOP includes the step of removing open frames.**]~~{.}~~

~~{57. A}~~

[57. (amended) The] method according to claim 51, wherein said splice-out point is determined in step (a) according to a user selection between an insert mode option and a splice-only mode option.

58. ~~{(amended)A}~~**[(twice amended) The]** method according to claim 57, wherein said splice-out point is determined as being immediately prior to a sequence header.

59. ~~{(amended)A}~~**[(twice amended) The]** method according to claim 57, wherein said splice-out point is determined as being immediately prior to a first occurring one of a group of pictures GOP header, an I-frame and a P-frame.

60. ~~{{(amended) A}}~~ **[(twice amended) The]** method according to claim 51, wherein said step of determining said splice-in point comprises:

finding a decode time stamp ("DTS") for a frame of the new data stream, said frame being included within a group of pictures GOP of the new data stream;

finding a corresponding presentation time stamp for said frame of the new data stream;

and

if said frame of the new data stream is other than an I-frame, then closing said GOP.

61. ~~{{(amended) A}}~~ **[(twice amended) The]** method according to claim 60, wherein said frame is the initial frame of the new data stream.

62. ~~{A}~~ **[(amended) The]** method according to claim 60, wherein finding said DTS includes parsing a first portion of the new data stream for a first sequence header, and then further parsing said first portion for a last DTS before a first frame header.

64. ~~{{(amended) A}}~~ **[(twice amended) The]** method according to claim 63, wherein step (b) is accomplished by deleting another frame within said portion that precedes said independently decodable frame.

66. ~~{A}~~ **[(amended) The]** method according to claim ~~{63}~~ [65], wherein step (d) of modifying includes replacing increasing temporal reference values of remaining frames within said GOP with correspondingly increasing temporal reference values of frames deleted in step (c).

69. ~~{A}~~ **[(amended) The]** method according to claim ~~{67}~~ [68] wherein said step of finding is accomplished in real-time during splicing of said new data stream with said old data stream.

70. ~~{A}~~ **[(amended) The]** method according to claim 68 wherein said inter-frame delay equals 1001/30,000 seconds.

71. ~~{(A)}~~**[(twice) amended]** A method for aligning a splice-out portion of a digitally encoded old data stream with a splice-in portion of a digitally encoded new data stream, comprising the step of setting a start of receipt time of said new data stream at which, if said new data stream is transmitted, then said new data stream will begin to be received by a decoder in alignment with a decoding time for said splice-out portion of said old data stream, and wherein said step of setting includes:

[if, upon transmission of said old and new data streams, said new data stream would begin to be received by a decoder before the decoder would have received all of said splice-out portion, then setting a transmission delay parameter for said new data stream; and]

if said new data stream, upon transmission, would begin to be received by the decoder after the decoder has received all of said splice-out portion, then setting a transmission acceleration parameter for said new data stream.

73. ~~{(A)}~~**[(amended)]** **The]** method according to claim ~~{72}~~ **[71]** that further includes inserting a number of null packets corresponding with said delay parameter into said new data stream at a position such that said null packets will be transmitted substantially prior to other new data stream data, if said new data stream is transmitted.

74. ~~{(amended) A}~~**[(twice amended)]** **The]** method according to claim 73 wherein said number of null packets equals a number of data packets that, without inserting the null packets, would be received by the decoder before the decoder has received all of said splice-out portion, if the new data stream is transmitted.

76. ~~{(Amended) A}~~**[(Twice amended)]** **The]** method according to claim ~~{75}~~ **[71]** that further includes deleting a number of null packets corresponding with said acceleration parameter from a first transmitted portion of said new data stream, if said new data stream is transmitted.

77. ~~{(amended) A}~~**[(twice amended)]** **The]** method according to claim 76 wherein said number of null packets equals a number of data packets that, without said deleting, would be received by the decoder after the decoder has received all of said splice-out portion, if the new data stream is transmitted.

80. ~~{(amended) An}~~ [(twice amended) **The**] apparatus according to claim 79, wherein said data stream portions include a new data stream portion with new data stream data and said new data stream data is received as a plurality of data packets.

81. ~~{An}~~ [(amended) **The**] apparatus according to claim 80, wherein said amount is calculated as a time corresponding to a number of data packets of data stream data.

~~{A}~~

[83. (amended) **The**] method according to claim 51 wherein the step of modifying the portion of the old data stream includes the step of dropping packets that exist in the old data stream.